



# **VIRTUALTAMPERE- MOBIILISOVELLUKSEN SUUNNITTELU JA TOTEUTUS IOS-ALUSTALLE**

Juho Järvensivu

Opinnäytetyö  
Marraskuu 2014  
Tietojenkäsittely  
Digimedia

TAMPEREEN AMMATTIKORKEAKOULU  
Tampere University of Applied Sciences

## TIIVISTELMÄ

Tampereen ammattikorkeakoulu  
Tietojenkäsittelyn koulutusohjelma  
Digimedia

JÄRVENSIVU, JUHO:

VirtualTampere-mobiilisovelluksen suunnittelu ja toteutus iOS-alustalle

Opinnäytetyö 30 sivua  
Marraskuu 2014

---

Opinnäytetyössä käsiteltiin VirtualTampere-verkkopalvelua. Se koostuu selaimessa toimivista 360°-panoraamaesityksistä, jotka on kuvattu Tampereen kiinnostavimmista kohteista. VirtualTampere-verkkopalvelun on toteuttanut tamperelainen JJ-Net Group Oy, joka toimii myös opinnäytetyön toimeksiantajana.

Opinnäytetyön tarkoituksena oli suunnitella ja toteuttaa VirtualTampere-verkkosivustosta uusi älypuhelimille optimoitu versio. Sitä hyödyntäen VirtualTampere-verkkopalvelulle toteutettiin myös oma iOS-mobiilisovellus. Mobiilisovelluksen toteutuksessa käytettiin hybriditekniikkaa. Tavoitteena oli kerätä tietoa erilaisten mobiilisovellusten tekemiseen liittyvistä tekniikoista, toteuttaa hyvin toimiva iOS-mobiilisovellus ja näin parantaa VirtualTampere-verkkopalvelun käytettävyyttä ja laajentaa sen potentiaalista käyttäjäkuntaa.

Opinnäytetyönä toteutettu uusi mobiilisivusto on osoittautunut toimivaksi, ja toimeksiantaja on ollut siihen tyytyväinen. Myös mobiilisovelluksen toteutus onnistui tavoitteiden mukaisesti, mutta sen julkaisu ei sisältynyt opinnäytetyöhön. Mobiilisovellusta kehitetään vielä opinnäytetyön valmistumisen jälkeen tekemällä käytettävyydestä. Mobiilisovellus julkaistaan ja mahdolliset versiot muille käyttöjärjestelmille toteutetaan myöhemmin.

---

Asiasanat: mobiilisovellus, responsiivisuus, hybridisovellus, sovelluskehys

## **ABSTRACT**

Tampereen ammattikorkeakoulu  
Tampere University of Applied Sciences  
Degree Programme in Business Information Systems  
Option of Digital Media

JÄRVENSIVU, JUHO:

Designing and Developing a VirtualTampere Mobile Application for iOS

Bachelor's thesis 30 pages

November 2014

---

The subject of this bachelor's thesis is VirtualTampere web service. VirtualTampere consists of panoramic image presentations that portray the most interesting attractions in Tampere. VirtualTampere is created by JJ-Net Group Oy which is also the client of the thesis.

The aim of this bachelor's thesis was to design and create a new mobile optimized version of the VirtualTampere website and to create a VirtualTampere iOS application. The mobile application is based on the new version of the VirtualTampere website and it was created as a hybrid app. The objective was to find out how different mobile app techniques differ from each other and then use that knowledge to create a functional iOS application. The application would enhance the usability of the VirtualTampere web service and make it accessible to a wider audience which was also a part of the objective.

The new mobile optimized website has been proven functional and client has been pleased with it. Also the development of the mobile application was completed as planned but its release wasn't included in the thesis. The further development of the mobile application will continue after the thesis with usability testing. The release of the mobile application and possible alternative versions for other operating systems will be made in the future.

---

Key words: mobile application, responsive web design, hybrid application, framework

## SISÄLLYS

1	JOHDANTO.....	6
2	OPINNÄYTETYÖN TAUSTA .....	8
2.1	Toimeksiantajan esittely .....	8
2.2	VirtualTampere .....	8
2.2.1	Tietoa panoraamakuvista .....	8
2.2.2	VirtualTampere-verkkopalvelun historia .....	10
2.2.3	VirtualTampere-vekkopalvelun uudistaminen.....	11
3	VIRTUALTAMPERE-VERKKOSIVUSTON MOBIILIVERSIO .....	12
3.1	Suunnitelu .....	12
3.1.1	Käyttöliittymä .....	12
3.1.2	Aloituskäyttö ja alisivut.....	13
3.2	Toteutus .....	16
3.2.1	Rakenne ja sisällöt.....	17
3.2.2	AJAX-tekniikan hyödyntäminen mobiiliversiossa .....	17
3.2.3	Käyttäjän päätelaitteen tunnistus.....	18
3.2.4	Responsiivisuuden toteutus .....	19
3.3	Mobiilisivuston testaus .....	21
4	VIRTUALTAMPERE-MOBIILISOVELLUS .....	22
4.1	Mobiilisovellustekniikan valitseminen .....	22
4.2	Hybriditekniikka VirtualTampere-mobiilisovelluksen toteutuksessa .....	24
4.3	Sovelluksen tekninen toteutus .....	25
4.4	Sovelluksen testaus .....	26
5	POHDINTA.....	29
	LÄHTEET .....	30

**ERITYISSANASTO**

AJAX	Koostuu useasta web-tekniikasta, joiden avulla internet-sivun sisältöä voidaan päivittää ilman koko verkkosivun uudelleenlatausta.
Apache	Avoimeen lähdekoodiin perustuva HTTP-palvelinohjelma.
Back-end	Sovelluskehitystermi, joka tarkoittaa muun muassa tietokantoja ja palvelimiin liittyviä asioita, jotka eivät ole loppukäyttäjän nähtävissä.
CSS	Cascading Style Sheets. HTML-sivuissa käytettävät tyyliohjeet.
Front-end	Sovelluskehitystermi, joka tarkoittaa loppukäyttäjän nähtävissä olevia asioita, kuten sovelluksen rakennetta ja sisältöjä.
HTML	Hypertext Markup Language. Internet-sivustoissa käytetty hypertekstin merkitäkieli, jolla merkitään sivun sisällöt ja rakenne.
iOS	Applen mobiililaitteiden käyttöjärjestelmä.
JavaScript	Dynaaminen komentosarjakieli, joka mahdollistaa sisällön interaktiivisen muokkaamisen komentosarjojen mukaan ilman verkkosivun uudelleenlatausta.
jQuery	Internet-selaimille tarkoitettu JavaScript-kirjasto, jonka suosio perustuu helposti ymmärrettävään syntaksiin.
Natiivisovellus	Sovellus, joka on kehitetty tietylle käyttöjärjestelmälle käyttäen sen omaa ohjelmointikieltä.
Objective-C	Ohjelmointikieli, jota käytetään muun muassa Applen iOS-käyttöjärjestelmässä.
Responsiivisuus	Mobiilioptimointi, eli käyttöliittymän mukautuvuus toimimaan optimaalisesti päätelaitteen näytön mukaan.
Sovelluskehys	Ohjelmisto, jota käytetään apuvälineenä helpottamaan ja nopeuttamaan sovelluksen tekemistä. Sovelluskehyksessä on ennalta määritelty runko toteutettavaa sovellusta varten, jota käyttäen se muodostaa valmiin sovelluksen ilman, että käyttäjän tarvitsee itse tehdä ohjelmointia.

## 1 JOHDANTO

Internetin selaaminen mobiililaitteilla on yleistynyt merkittävästi 2000-luvun aikana. Lähtölaukaus mobiililaitteiden yleistymiselle tapahtui vuonna 2007, kun Apple julkaisi suureen suosioon nousseen iPhone-älypuhelimien. Wroblewskin (2011, 10) mukaan iPhone oli mullistava, koska se oli ensimmäinen älypuhelin, jolla internetin selaaminen oli luontevaa. Jo vuonna 2009 yksi iPhone käytti keskimäärin yhtä paljon mobiilidata-liikennettä kuin 30 tavallista puhelinta. Älypuhelimien yleistymiseen vaikuttivat myös niiden halvemmat hinnat verrattuna pöytäkoneisiin ja kannettaviin tietokoneisiin. Tietokoneen ja laajakaistayhteyden omistaminen ei enää ollut vaatimus internetiin pääsemiseen, vaan siihen riitti älypuhelin ja mobiilidatapaketti, joiden matalampi hintataso toi ne suuremman yleisön saataville. (Wroblewski 2011, 10 - 11.)

Mobiililaitteiden suurten käyttäjämäärien takia internet-sivustojen ja -palveluiden toteutuksessa täytyy nykyään ottaa huomioon käytettävyys mobiililaitteilla, ja nettisivujen responsiivisuus onkin nykypäivänä enemmän sääntö kuin poikkeus. Verkkosivustojen tai -sovellusten mobiililähtöinen suunnittelu ei ole pelkästään potentiaalinen tapa kasvattaa kävijävirtoja, vaan se voi lisäksi parantaa olennaisesti palveluiden käyttökokemusta (Wroblewski 2011, 1). Monet suositut verkkopalvelut, kuten Facebook ja YouTube, tarjoavat käyttäjilleen erillisiä mobiilisovelluksia vaihtoehtoisena tapana käyttää verkkopalveluitaan. Mobiilisovellusten käyttämisestä on tullut erittäin suosittua: tammikuussa 2014 internetiä käytettiin Yhdysvalloissa ensimmäistä kertaa enemmän mobiilisovellusten kautta kuin pöytäkoneilla (O'Toole, 2014).

Tässä opinnäytetyössä suunnitellaan ja toteutetaan VirtualTampere-verkkosivustosta mobiililaitteille optimoitu versio ja tehdään sitä hyödyntäen VirtualTampere-palvelulle oma mobiilisovellus iOS-alustalle. Opinnäytetyössä kerrotaan, mitä asioita pitää huomioida verkkosivuston mobiilioptimointia tehtäessä, minkälaista toteutustekniikkaa kannattaa käyttää erilaisten mobiilisovellusten toteutuksessa ja kuinka verkkosivustosta pystyy toteuttamaan mobiiliapplikaation käyttäen sovelluskehystä.

Opinnäytetyöprojektia aloittaessaan työn tekijällä on jo valmiiksi kokemusta responsiivisten verkkosivustojen suunnittelusta ja toteutuksesta, mutta mobiilisovelluksen tekemisestä hänellä on vain teoretietoa. Tavoitteena on kerätä tietoa erilaisten mobiilisovel-

lusten tekemiseen liittyvistä tekniikoista ja sitä hyödyntäen toteuttaa hyvin toimiva mobiilisovellus iOS-alustalle. Opinnäytetyössä pääpaino on suunnittelussa ja toteutuksesta kuvataan pääpiirteitä.

## **2 OPINNÄYTETYÖN TAUSTA**

### **2.1 Toimeksiantajan esittely**

JJ-Net Group Oy on tamperelainen ICT-alan yritys, joka on perustettu vuonna 2003. Yritys tuottaa palveluita internetiin ja näistä lähes kaikki käyttävät alustanaan JJ-Net Group Oy:n omaa TSI24-sisällönhallintajärjestelmää. Opinnäytetyön kirjoitushetkellä, syksyllä 2014, JJ-Net Group Oy työllistää 12 työntekijää. JJ-Net Group Oy:n tuottamiin palveluihin kuuluvat internet-sivustot, verkkokaupat, 360°-panoraamaesitykset sekä monet muut internetiin toteutetut palvelut. (JJ-Net Group Oy 2014.)

### **2.2 VirtualTampere**

VirtualTampere on JJ-Net Group Oy:n toteuttama ilmainen verkkopalvelu, jonne on koostettu panoraamakuvaesityksiä Tampereen kiinnostavimmista nähtävyyksistä ja palveluista. VirtualTampere-verkkosivusto on avattu vuonna 2011. (VirtualTampere 2014.)

#### **2.2.1 Tietoa panoraamakuvista**

Panoraamakuvaesitykset ovat tavanomaisesti internet-selaimessa toimivia 360°-kuvaesityksiä, joiden avulla käyttäjä pystyy kokemaan esitetyn kohteen, kuin hän olisi itse paikan päällä. Tunnetuin esimerkki panoraamakuvapalvelusta on Google Street View -katunäkymäpalvelu.

Perinteinen 360°-panoraamakuva on useimmiten koostettu useasta erillisestä valokuvasta, jotka on jälkikäteen yhdistetty toisiinsa, niin että ne yhdessä muodostavat yhden 360°-valokuvan. Valmis panoraamakuva on nähtävissä kuvassa 1. Tämä valokuva upotetaan internet-sivulle panoraamaesityssovelluksen kanssa, minkä jälkeen käyttäjä pystyy internet-selaimessa pyörittämään valittua panoraamanäkymää upotetun käyttöliittymän kautta.





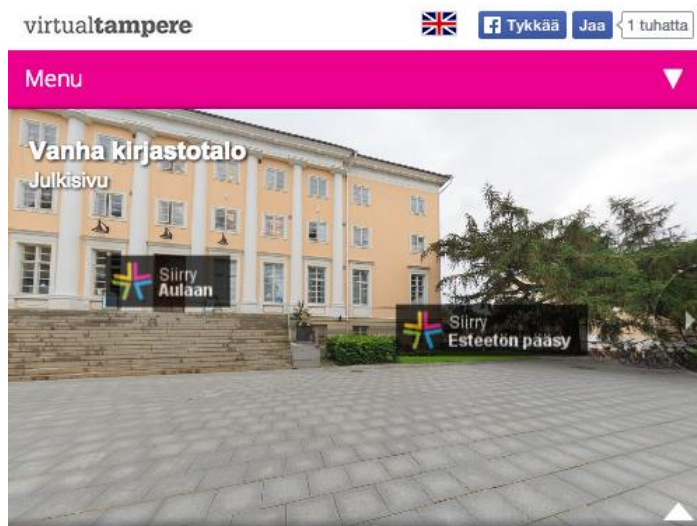
KUVA 1. Valmiiksi käsitelty suorakaiteenmuotoinen panoraamakuva VirtualTampere-palvelusta. Kuva on Tampereen ortodoksisesta kirkosta. (JJ-Net Group Oy 2014.)

Panoraamaesityksen muodostavat yksittäiset valokuvat otetaan yleensä niin, että kamera sijoitetaan jalustalle ja kuvataan erilliset valokuvat jalustan pysyessä paikalla. Kuvien ottamisen välillä kameraa pyöritetään horisontaalisesti esimerkiksi 90 astetta akselinsa ympäri. Näin kohde saadaan kuvattua yhdestä pisteestä neljään eri suuntaan ja näistä kuvista pystytään jälkikäteen koostamaan yksi 360°-valokuva. Panoraamakuvausta varten kamerassa käytetään yleensä kalansilmäobjektiivia, jotta yksittäisen kuvan kattavuus koko panoraamasta on mahdollisimman suuri. Panoraamaesitysovelluksessa valmiit 360°-panoraamakuvat esitetään pallomuodossa, jolloin käyttäjän katselupiste on pallon keskipisteessä ja hän pystyy panoraamasovelluksen käyttöliittymän avulla vaihtelevaan katselusuuntaansa. Havainnollistava kuvaesimerkki pallomuodossa esitetystä panoraamakuvausta on nähtävissä kuvassa 2.



KUVA 2. Pallomuodossa esitetty panoraamakuva (Street View Service 2014).

Jos samasta ympäristöstä otetaan useita 360°-panoraamakuvia eri katselupisteistä, voidaan näistä koostaa yksittäinen panoraamaympäristö. Panoraamaympäristössä näihin kuviin usein lisätään erilaisia hotspot-painikkeita, joiden avulla käyttäjä voi liikkua panoraamakuvasta toiseen. Esimerkki panoraamaesityksen hotspot-painikkeista on nähtävissä kuvassa 3.



KUVA 3. Kaksi hotspot-painiketta VirtualTampereen panoraamaesityksessä.

### 2.2.2 VirtualTampere-verkkopalvelun historia

JJ-Net Group Oy on toteuttanut panoraamaesityksiä vuodesta 2003 alkaen. Yritys päätti kehittää ilmaisen VirtualTampere-panoraamaesityspalvelun, josta tuli kanava yrityksen panoraamaosaamisen esittelyyn, sillä JJ-Net Group Oy:n pankki- ja teollisuusyrityksille toteuttamia panoraamaratkaisuja ei saanut esitellä julkisesti. VirtualTampere on yritykselle markkinointiväylä, jossa se pystyy lisäksi vapaasti testaamaan kaikenlaisten uusien tekniikoiden toimivuutta käytännössä. (Jeskanen 2014.)

Kosketusnäytöillä varustettujen älypuhelinien ja tablet-tietokoneiden yleistyttyä ja niiden teknisten ominaisuuksien kehittyttyä riittäväälle tasolle huomattiin, että kosketusnäytöllä varustettu laite onkin optimaalinen päätelaite 360°-panoraamaesitysten selaamiseen. Erityisesti tablet-tietokoneille sai tehtyä näyttäviä panoraamaesityksiä muun muassa messukäyttöön, koska tablet-tietokoneen suuri kosketusnäyttö teki panoraamaesitysten selaamisesta sujuvaa. Koko näytön pinta-alalle asemoitu panoraamaesitys on

myös erittäin näyttävä visuaalisesti. Kosketusnäyttölaitteiden yleistyminen on tehnyt panoraamaesitysten käyttökokemuksesta intuitiivisemman, ja se on ylipäättään mahdollistanut panoraamaesitysten käyttämisen mobiilisti. (Jeskanen 2014.)

### **2.2.3 VirtualTampere-vekkopalvelun uudistaminen**

Vuoden 2014 alussa koko VirtualTampere-verkkosivusto uudistettiin. Lähtökohtana siinä oli optimaalinen toimivuus kosketusnäytöllisillä laitteilla, ensisijaisesti tablet-tietokoneilla. Uudistettu VirtualTampere julkaistiin keväällä 2014. Uudistettu verkkosivusto oli tehty kokonaan responsiiviseksi, eli sivuston elementit skaalautuivat päätelaitteen näytön mukaan. Tästä huolimatta sivuston käytettävyys älypuhelimilla ei ollut toivotulla tasolla. Tablet-tietokoneille suunniteltu käyttöliittymä vei liian suuren tilan älypuhelimien pienestä näytöstä, eikä sivustolla olleita elementtejä saatu aina asemoitua älypuhelimien näytölle luonteviin paikkoihin, vaan käyttäjän piti usein vierittää sivua alaspäin nähdäkseen kaikki sivun sisällöt. Tästä johtuen kesällä 2014 alettiin suunnitella ja toteuttaa VirtualTampere-sivustosta uutta erityisesti älypuhelimille optimoitua versiota. (Jeskanen 2014.)

### 3 VIRTUALTAMPERE-VERKKOSIVUSTON MOBIILIVERSIO

#### 3.1 Suunnitelu

VirtualTampere-palvelun mobiilikäyttöliittymän suunnittelussa lähdettiin liikkeelle puhtaalta pöydältä ja hylättiin kokonaan olemassa olevan verkkopalvelun käyttöliittymä. Lähtökohtana suunnittelulle oli se, että itse käyttöliittymä veisi näytöltä mahdollisimman vähän tilaa kuitenkin niin, että VirtualTampere-logo olisi koko ajan näkyvissä. Tämän lisäksi näytön alareunaan laitettaisiin linkki JJ-Net Group Oy:n kotisivulle sekä Facebookin jakopainikkeet. (Mattus 2014.)

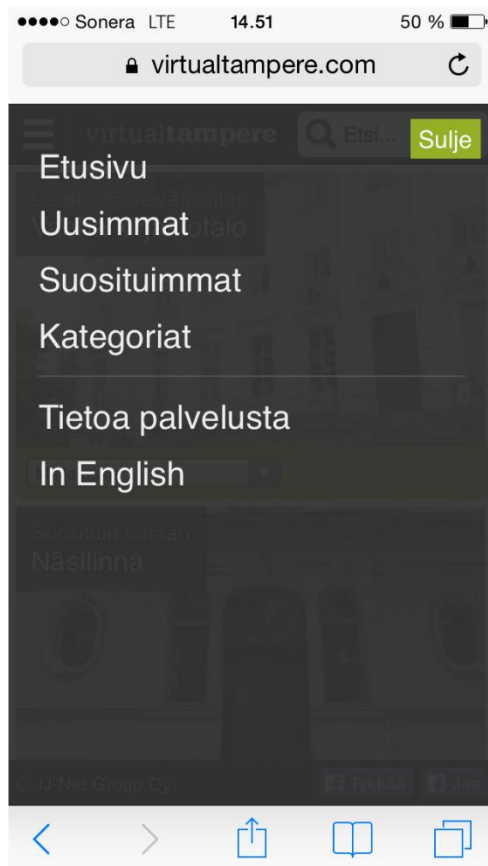
Ongelma aiemmassa käyttöliittymässä mobiililaitteilla oli se, että panoraamaesityksen lisätietokentät oli sijoitettu itse panoraamaesityksen alapuolelle ja ne saattoivat jäädä käyttäjältä kokonaan huomaamatta, sillä sivua piti vierittää alaspäin, että nämä kentät tulivat näkyviin pienemmillä näytöillä. Uuden käyttöliittymän suunnittelussa pyrittiin siihen, että kaikki sisältö olisi käyttäjän nähtävillä ilman, että sivua täytyisi vierittää. Itse panoraamaesitykset olisivat käyttöliittymän suunnittelussa etusijalla, ja niille varattaisiin näytöltä niin paljon pinta-alaa kuin mahdollista. (Mattus 2014.)

##### 3.1.1 Käyttöliittymä

Näytön yläreunaan päätettiin laittaa käyttöliittymäpalkki, jossa on VirtualTampere-logo, hakupalkki sekä ikoni, jota klikkaamalla valikko tulee näkyviin. Käyttöliittymäpalkkiin laitettiin musta taustaväri, jotta mobiilikäyttöliittymä eroaisi selkeästi suuremmille näytöille tehdystä versiosta. Käyttöliittymäpalkki määriteltiin pysymään näkyvissä näytön yläreunassa koko ajan.

Käyttöliittymäpalkista avattavaan valikkoon laitettiin linkki VirtualTampere-sivuston etusivulle sekä muutama linkki, joiden avulla käyttäjä pystyy etsimään panoraamaesityksiä erilaisia suodattimia käyttäen (uusimmat, suosituimmat, kategoriat). ”Tietoa palvelusta” -linkkiä klikkaamalla pääsee lukemaan lyhyen esittelytekstin VirtualTampere-verkkopalvelusta. Näiden lisäksi valikkoon laitettiin ”In English/Suomeksi”-linkki kie-

liversioon vaihtamiseen sekä ”Sulje”-painike valikon sulkemista varten. Kuvakaappaus uudistetun käyttöliittymän valikosta on nähtävissä kuvassa 4.



KUVA 4. Kuvakaappaus mobiilisivuston uudistetusta valikosta iPhone 5S –älypuhelimella.

### 3.1.2 Aloituskäyttö ja alisivut

Aloituskäyttöön laitettiin suoraan valittaviksi kaksi panoraamaesitystä, jotka ovat uusien palveluun lisätty panoraamaesitys sekä päivän suosituin panoraamaesitys. Näiden yhteyteen lisättiin myös pudotusvalikot, joissa on valittavana kahdeksan seuraavaksi uusinta tai suosituinta kohdetta. Näiden panoraamaesitysvalintojen alapuolelle lisättiin lyhyt VirtualTampere-palvelun kuvausteksti. Kuvakaappaus uudistetusta käyttöliittymästä on nähtävissä kuvassa 5.



KUVA 5. VirtualTampere-verkkopalvelun uudistetun etusivun kuvakaappaus iPhone 5S –älypuhelimella.

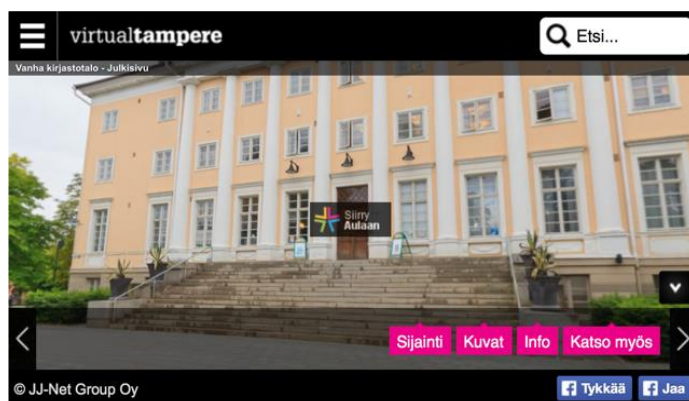
Valikon ”Uusimmat”- ja ”Suosituimmat”-linkeistä avautuvat sivut määriteltiin toimimaan keskenään samanlaisella asettelulla ja näille sivuille tulee näkyviin kuusi panoramaesitystä suoraan valittaviksi. ”Kategoriat”-sivu toimii eri tavalla, ja siinä aluksi tulee näkyviin vain pudotusvalikko, josta on haettavissa panoramaesityksiä seuraavissa kategorioissa:

- suosittelemme
- nähtävyydet
- majoitus
- aktiviteetit
- kulttuuri
- liikenne
- palvelut
- kaikki.

Kun käyttäjä valitsee kategorian, tulevat näkyviin valintapainikkeet kaikkiin panoraamaesityksiin, jotka kuuluvat valittuun kategoriaan. Valintasivuilla olevat painikkeet päätettiin pitää riittävän suurina, että niitä on helppo klikata, mutta kuitenkin niin, että niitä mahtuisi näytölle noin kuusi kappaletta kerrallaan.

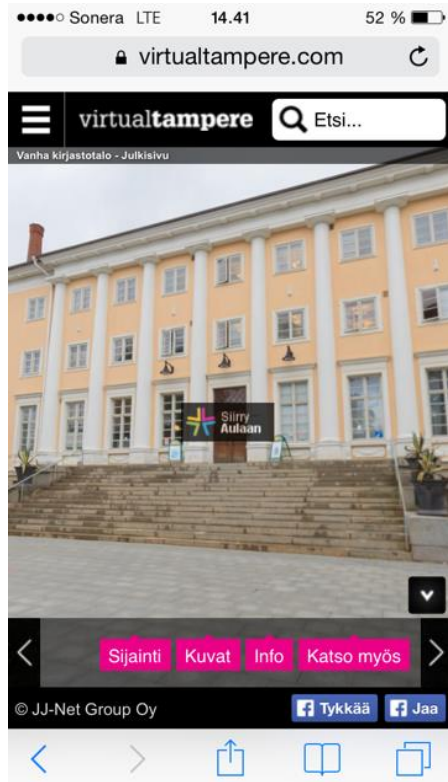
Itse panoraamasivulla esitys asemoituu automaattisesti koko näytön korkuiseksi ja leveyiseksi. Panoraaman yläreunan päällä lukee läpinäkyvällä taustalla esityksen sekä valitun kuvan nimi. Panoraamaesityksen alareunaan laitettiin valikkopalkki, jonka reunoista löytyy nuolet edellisen tai seuraavan panoraamakuvan valitsemiseen. Alavalikon päällä oikeassa reunassa on pieni painike, jonka avulla alavalikon pystyy halutessaan piilottamaan tai tuomaan takaisin näkyviin. Panoraamaesityksen lisätietokentät löytyvät alavalikossa olevia linkkipainikkeita klikkaamalla. Valittu lisäsisältö tulee näkyviin panoraamaesityksen päälle liukuvalla animaatiolla.

Panoraamasivun ensimmäinen lisätietokenttä on ”sijainti”, jota klikkaamalla tulee näkymiin valitun kohteen sijainti kartalla, joka esitetään Google Maps -karttapalvelun avulla. Toinen lisätietopainike on ”kuvat”, jota klikkaamalla näytölle aukeaa lista valitun panoraamaesityksen mahdollisista toisista panoraamakuvista. Käyttäjä pystyy siirtymään toiseen panoraamakuvaan klikkaamalla linkkiä listalta. Kolmas lisätietopainike ”info” sisältää sanallisen kuvauksen valitusta panoraamaesityksestä. Viimeinen lisätietopainike on ”katso myös” ja tämän painikkeen kautta tulee näkyviin lista saman kategorian kohteista, joita käyttäjä voi siirtyä katsomaan valitsemalla linkin aukeavasta listasta. Uudistettu panoraamasivun näytön mukaan skaalautuva käyttöliittymä on nähtävissä kuvissa 6 ja 7.



KUVA 6. Kuvakaappaus uudistetusta panoraamasivusta iPhone 5S -älypuhelimella vaaka-asennossa.





KUVA 7. Kuvakaappaus uudistetusta panoraamasivusta iPhone 5S -älypuhelimella pystyasennossa.

### 3.2 Toteutus

VirtualTampere-verkkosivuston uuden mobiiliversion toteuttamista aloitettaessa verkkosivuston keväällä 2014 tehty responsiivinen versio oli julkaistu ja täysin toiminnassa. Jotta VirtualTampere-verkkosivuston tulevaa mobiiliversiota pystyttiin työstämään verkkosivuston julkaistun version rinnalla, määriteltiin kehitteillä oleva mobiiliversio aukeamaan vain, mikäli selaimen osoiteriville oli URL-osoitteen perään kirjoitettu ennalta määritelty testitunniste ”mobiledebug”. Selain tarkistaa sivulatauksen yhteydessä käyttäjän syöttämän URL-osoitteen ja tämän perusteella näytetään oikea versio sivustosta. Tällä tavoin mobiiliversiota pystyttiin rauhassa kehittämään autenttisissa olosuhteissa jo julkaistun version rinnalla. Projektin tekemiseen osallistui työkollega JJ-Net Group Oy:stä ja hän vastasi projektissa back-end-toteutuksesta. (Mattus 2014.)



### **3.2.1 Rakenne ja sisällöt**

HTML-rakenne tehtiin kaikkiin uuden mobiiliversion sivunäkymiin alusta asti uudestaan, mutta onneksi muissa työvaiheissa pystyttiin hyödyntämään alkuperäiselle sivustolle toteutettuja tekniikoita. Esimerkiksi panoraamojen esityssovellus on sama kuin julkaistussa versiossa ja panoraamaesitysten haku- ja suodatusominaisuuksissa käytetään samoja tekniikoita kuin VirtualTampereen alkuperäisillä verkkosivuilla. (Mattus 2014.)

VirtualTampere-sivuston sivurakenne on toteutettu normaalista poikkeavalla tavalla verrattuna tavallisiin verkkosivustoihin. Sen sivurakenne sisältää ainoastaan etusivun ja sivuston kaikki muut sivut generoidaan näkyviin dynaamisesti selaimen osoiterivillä olevan sivupolun perusteella. Panoraamaesitysten käyttämässä hallintatyökalussa on useita tietokenttiä esityssivulle tulevia tietoja varten, sekä kentät, joiden kautta jokaiselle esitykselle ja yksittäisille panoraamoille määritellään uniikit sivupolut. Kun selain suorittaa kyselyn sellaiselle sivulle, jota ei ole olemassa, ohjataan tehty kysely sellaiseen etusivulle Apachen rewrite-moduulia käyttäen. Siellä kyselyssä olleen sivupolun perusteella haetaan tietokannasta olemassaolevia esityksiä ja panoraamoja. (Mattus 2014.)

### **3.2.2 AJAX-tekniikan hyödyntäminen mobiiliversiossa**

VirtualTampere-sivuston uusi versio eroaa alkuperäisestä sisältöjen esittämisessä käytettävän tekniikan osalta. Mobiilisivustolla sisältöjen päivittämisessä käytetään AJAX-tekniikkaa, eli aina kun siirrytään uudelle sivulle, avattavan sivun sisällöt haetaan palvelimelta ja päivitetään näkyviin ilman, että selaimen tarvitsee ladata koko verkkosivua uudestaan. AJAX-tekniikan ansiosta sivusto toimii nopeammin ja ladattavat datamäärät ovat pienemmät, sillä esimerkiksi panoraamaesityssovellusta ei tarvitse ladata uudestaan jokaista panoraamaesitystä varten. Sivuston nopeatoimisuus on mobiilisivustolla tärkeää, sillä sen pitäisi toimia kelvollisesti myös hitaammilla mobiilidatayhteyksillä. (Mattus 2014.)

Koska AJAX-tekniikkaa käyttämällä verkkosivua ei ladata uudelleen toiselle sivulle siirryttäessä, käytetään tässä avuksi HTML5:n History-ohjelmointirajapintaa. Kun mo-

biilisivustolle vaihdetaan sivun sisältöä AJAX-tekniikkaa käyttäen, pystytään History-ohjelmointirajapinnan avulla samalla päivittämään selaimen osoiterivillä oleva sivupolku näytetyn sivun mukaiseksi. Tämän lisäksi History-ohjelmointirajapinnan avulla selaimen ”edellinen sivu”- ja ”seuraava sivu”-painikkeet ovat käytettävissä normaaliin tapaan ja niitä klikkaamalla selain näennäisesti siirtyy edelliselle sivulle, kuten staattisesti toimivilla sivuilla. AJAX-tekniikan ja History-ohjelmointirajapinnan yhdistelmällä yhdessä polkutunnistuksen kanssa pystytään luomaan käyttäjälle illuusio aivan normaalisti toimivasta sivustosta, vaikka se sisältääkin ainoastaan yhden sivun, eikä suorita ollenkaan sivulatauksia. (Pilgrim 2014.)

### **3.2.3 Käyttäjän päätelaitteen tunnistus**

Kun verkkosivustosta oli toteutettuna kaksi erilaista versiota, piti käyttäjät pystyä ohjaamaan haluttuun versioon. Tämä onnistui user-agentin, eli selaimen tunnistetiedon avulla. Selaimen tunnistetiedon perusteella pystytään tunnistamaan, mikä selain ja päätelaite kävijällä on käytössä. Jos käytössä on mobiililaitteen selain, käyttäjä ohjataan uuteen mobiilioptimoituun versioon. Muussa tapauksessa näkyviin aukeaa alkuperäinen versio verkkosivuista. (Mattus 2014.)

Jotta uusi mobiiliversio saatiin näkymään mobiililaitteilla halutulla tavalla, piti HTML-sivuun lisätä vielä niin sanottu viewport-määrittely. Sen avulla mobiililaitteen selaimelle kerrotaan, kuinka selaimen tulee käyttäytyä kyseisen sivun kanssa. Viewport-määrittely on julkaistu vuonna 2007 Applen ensimmäisen iPhoneen yhteydessä, ja se on myöhemmin sisällytetty toimimaan myös muissa mobiiliselaimissa. (Marcotte 2011, 80.)

Vaikka ensimmäisen iPhoneen näytön todellinen koko oli 320 x 480 pikseliä, näytti iPhoneen Safari-selain sivustot oletuksena 980 pikselin levyisenä. Koska VirtualTampere-sivuston mobiiliversion haluttiin muistuttavan käytettävyydeltään mobiilisovellusta, määriteltiin selain käyttämään verkkosivun leveytenä puhelimen näytön todellista leveyttä viewportin avulla ilman oletus-zoomausta ja poistimme käyttäjältä mahdollisuuden itse zoomata sivua.

### 3.2.4 Responsiivisuuden toteutus

Ohjenuorina älypuhelimille optimoidun verkkosivuston responsiivisen ulkoasun suunnittelussa ja toteutuksessa olivat seuraavat asiat:

- Koko päätelaitteen näytön pinta-ala pitää hyödyntää mahdollisimman hyvin.
- Sivustolla olevien elementtien pitää olla riittävän suuria ja kuvien sekä tekstien täytyy pysyä hyvin luettavina, kuitenkin niin, ettei mikään yksittäinen elementti vie näytöltä kohtuuttomasti tilaa.
- Käytettävyyden pitää olla hyvä vielä 320 pikseliä leveällä laitteella, koska tämä on iPhone-älypuhelimien selaimen leveys puhelimen ollessa pystyasennossa.

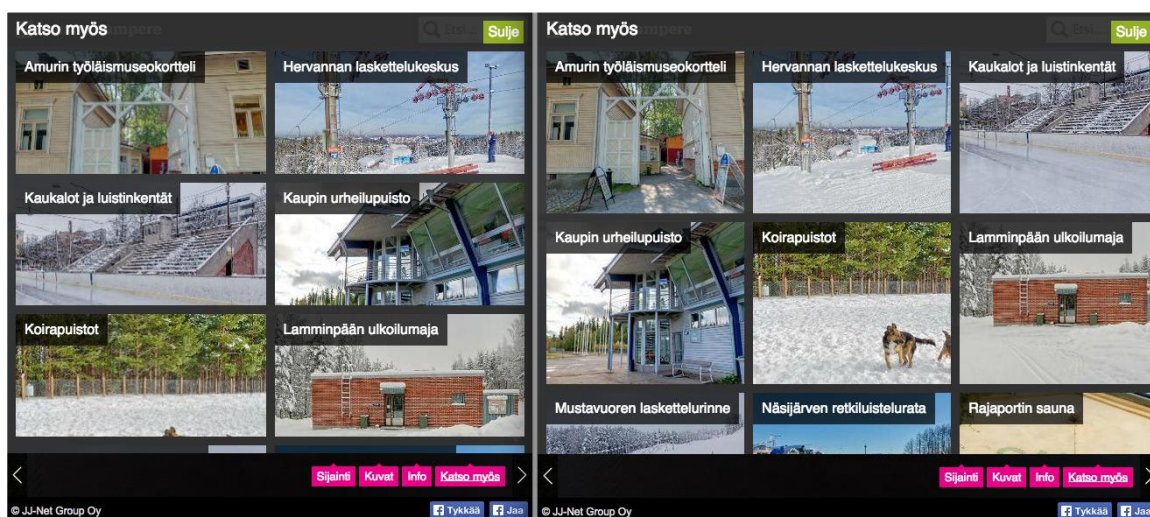
Ethan Marcotte mainitsee kirjassaan ”Responsive Web Design”, että responsiivinen web-suunnittelu koostuu kolmesta ydinasiasta, jotka ovat (Marcotte 2011, 9.)

- mukautuva, asetteluruudukkoon pohjautuva ulkoasu (a flexible, grid-based layout)
- mukautuvat kuvat ja mediasisällöt (flexible images and media)
- CSS-tyyliohjeiden mediakyselyt (media queries).

Marcotten mainitsemaa asetteluruudukkoa voidaan käyttää sivuston ulkoasun suunnittelussa seuraavasti: Sivuston sisältöalue jaetaan esimerkiksi 12:een yhtä suureen osaan ja näitä osia hyödyntäen muodostetaan sisältöalueesta ruudukko. Sivulle tulevat HTML-elementit asetellaan sisältöalueelle käyttäen asetteluruudukkoa, jotta sivun elementtien asettelu saadaan symmetriseksi. Asetteluruudukon avulla sivun elementeille voidaan laskea prosentuaaliset leveydet, jolloin sivuston elementit saadaan skaalautumaan leveysuunnassa selainikkunan koon mukaan.

VirtualTampere-sivuston 2014 keväällä uudistetun version ulkoasu oli toteutettu mukautuvaksi hyödyntäen asetteluruudukkoa, mutta opinnäytetyössä toteutettavan version suunnittelussa ei asetteluruudukkoa käytetty juurikaan hyödyksi. Tämä johtui siitä, että mobiiliversiossa lähes jokainen elementti on määritelty joko asettumaan absoluuttisesti tiettyyn kohtaan selainikkunaa tai vaihtoehtoisesti skaalautumaan aina koko selainikkunan levyiseksi. Toisin kuin tavallisilla verkkosivustoilla, VirtualTampere-sivuston mobiiliversiossa sisältöalueelle ei ole määritelty maksimileveyttä, vaan sisältöalue asemoi-  
tuu aina koko selainikkunan levyiseksi.

VirtualTampereen mobiiliversion ainoat vierekkäin asettuvat, skaalautuvat elementit ovat päävalikon kategoriasuodatusten tai panoraamaesityssivun ”kuvat” ja ”katso myös” -linkkien kautta löytyvät panoraamaesitysten valintapainikkeet. Edellä mainitut elementit asettuvat joko kahden tai kolmen elementin riveihin selainikkunan koosta riippuen, kuten on nähtävissä kuvasta 8. Näille valintapainikkeille on määritetty taustakuviksi aina kyseisestä kohteesta luotu esikatselukuva. Taustan esikatselukuvat saatiin toimimaan mukautuvasti lisäämällä elementeille CSS3-tyyliohje: ”background-size:cover;”. Tämän määrittelyn avulla taustakuva asemoituu aina optimaalisesti, vaikka elementin mittasuhteet vaihtuvatkin. Valintapainikkeita lukuun ottamatta sivustolla ei ole muita vierekkäin asemoituja elementtejä, joiden suunnittelussa olisi voitu hyödyntää asetteluruudukkoa. Tästä johtuen VirtualTampere-sivuston mobiiliversion ulkoasu on mukautuva, mutta se ei suoranaisesti pohjautu asetteluruudukkoon.



KUVA 8. ”Katso myös” -osion valintapainikkeet asettuvat kolmen vierekkäisen elementin riveihin, kun selainikkunan leveys on yli 680 pikseliä.

VirtualTampere-sivuston mobiiliversionissa olevat kuvat ja mediasisällöt on määritetty mukautumaan päätelaitteen näytön koon mukaan. Tosin sivuston ainoa tavallinen HTML-kuvaelementti on yläreunassa oleva VirtualTampere-logo. Logolle määriteltiin etukäteen sellaiset mitat, että se sopii hyvin erikokoisille näytöille ilman, että sitä täytyy skaalata. Sivuston muut mediasisällöt ovat panoraamasivun panoraamaesitys ja sivustolta löytyvät Google Maps -kartat. Molemmat edellä mainituista mediasisällöistä määriteltiin asettumaan maksimileveyteen ja -korkeuteen päätelaitteen näytön pinta-alan ja sivun rakenteen asettamissa rajoissa käyttäen jQuerya.

Marcotten kolmas ydinasia oli CSS-tyyliohjeiden mediakyselyt. Mediakyselyiden avulla elementeille pystytään määrittelemään vaihtoehtoisia CSS-tyyliohjeita internet-selainikkunan koon perusteella. Edellä mainittiin tästä esimerkki kuvan yhteydessä, jossa panoraamaesitysten valintapainikkeiden leveyttä muokattiin, kun selainikkuna ylitti määrätyn leveyden. Mediakyselyä käyttäen valintapainikkeet saatiin tässä tapauksessa asemoitua kolmen elementin riveihin. Muualla sivuston tyylimäärittelyissä mediakyselyitä hyödynnettiin otsikkofonttien koon pienentämiseen ja elementtien madaltamiseen, kun selainikkuna oli riittävän kapea.

### **3.3 Mobiilisivuston testaus**

Mobiilisivuston testausvaiheessa havaittiin ongelmia pari vuotta vanhojen iOS-laitteiden kanssa. Ongelma esiintyi, kun panoraamaesitystä katseltaessa mobiililaitetta käännettiin pysty- ja vaakaorientaatioiden välillä ja avattiin samalla panoraamaesityksen valikoita. Tällaisissa tilanteissa mobiilisivusto kuormitti suhteellisen paljon keskusmuistia, mikä johti mobiiliselaimen kaatumiseen. Uudemmissa iOS-laitteilla samaa ongelmaa ei saatu toistettua, niiden suuremman keskusmuistin ansiosta. Ongelman esiintymistä saatiin kuitenkin harvennettua määrittelemällä mobiililaitteiden käyttöön resoluutioltaan matalammat panoraamaesitykset, jotka kuormittivat laitteen keskusmuistia huomattavasti vähemmän.

## 4 VIRTUALTAMPERE-MOBIILISOVELLUS

VirtualTampere-mobiilisivuston uusi versio paransi sen käytettävyyttä älypuhelimilla, mutta sen käytettävyyteen liittyi yksi perustavanlaatuinen ongelma, josta haluttiin päästä eroon. Tämä ongelma oli älypuhelimien internet-selainten omat käyttöliittymät, jotka veivät merkittävän osan päätelaitteiden näyttöjen pinta-alasta. Internet-selaimet itsessään saattoivat osoiteriveineen ja navigointipalkkeineen viedä pahimmillaan noin kolmasosan koko näytön pinta-alasta, mikä toi suuria haasteita mobiilikäyttöliittymän suunnitteluun. Visuaalisuuteen perustuvien panoraamaesitysten käyttökokemus kärsii pahasti liian pienestä näytöstä. Tästä saatiinkin idea toteuttaa VirtualTampere-palvelusta oma mobiilisovellus, joka toimisi muutoin samalla tavalla kuin VirtualTampere-palvelun verkkosivusto, mutta poistamalla näkyvistä puhelimen internet-selaimen oma käyttöliittymä saataisiin sovelluksen käyttöön koko päätelaitteen näytön pinta-ala. Oman sovelluksen tekeminen nähtiin myös mahdollisuutena kasvattaa VirtualTampereen käyttäjämäärää ja näkyvyyttä. (Jeskanen 2014.)

### 4.1 Mobiilisovellustekniikan valitseminen

Mobiilisovellukset voidaan teknisen toteutuksen näkökulmasta jakaa kolmeen kategoriin: natiivisovellukset, hybridisovellukset ja web-sovellukset. Kaikissa näissä toteutusmalleissa on hyvät ja huonot puolensa, ja parhaiten sopiva sovellustekniikka riippuukin siitä, millaista sovellusta ollaan tekemässä. Mobiilisovelluksessa käytettävää tekniikkaa valittaessa täytyy kiinnittää huomiota muun muassa seuraaviin asioihin:

- Mikä on mobiilisovelluksen tarkoitus, ja kenelle se on suunnattu?
- Tehdäänkö mobiilisovellusta useammalle kuin yhdelle käyttöjärjestelmälle?
- Jos mobiilisovellusta tehdään itse, millaista teknistä osaamista yrityksen työntekijöillä on?
- Jos mobiilisovellus tilataan ulkopuoliselta toimittajalta, kuinka suuri budjetti mobiilisovelluksen toteutukseen on?
- Millaisia teknisiä ominaisuuksia mobiilisovelluksessa käytetään? Tarvitaanko pääsyä esimerkiksi kameraan, paikannustietoihin?
- Kuinka usein ja millä tavoin sovellusta on tarkoitus päivittää?

Natiivisovellus toteutetaan jokaiselle käyttöjärjestelmälle erikseen, käyttäen kunkin käyttöjärjestelmän ohjelmointikieltä. Se on teknisiltä ominaisuuksiltaan kaikkein laajin vaihtoehto ja lisäksi myös nopeatoimisin. Jos halutaan, että natiivisovellus toimii usealla käyttöjärjestelmällä, pitää sovelluksen eri versiot toteuttaa alusta asti uudestaan jokaiselle käyttöjärjestelmälle erikseen, vaikka periaatteessa haluttaisiin vain yksi sovellus, joka on tuettu usealla käyttöjärjestelmällä. Natiivisovellus voi siis osoittautua kalliiksi tai työlääksi vaihtoehdoksi, koska työmäärä on kerrannainen tuettujen käyttöjärjestelmien määrällä ja lisäksi jokainen sovellus vaatii eri ohjelmointikielten osaamista. Mikäli budjetti tai työntekijöiden ohjelmointitaidot ja resurssit ei ole kynnyskysymys, on natiivisovellus yleensä lopputuloksen kannalta paras vaihtoehto. (Budi 2013.)

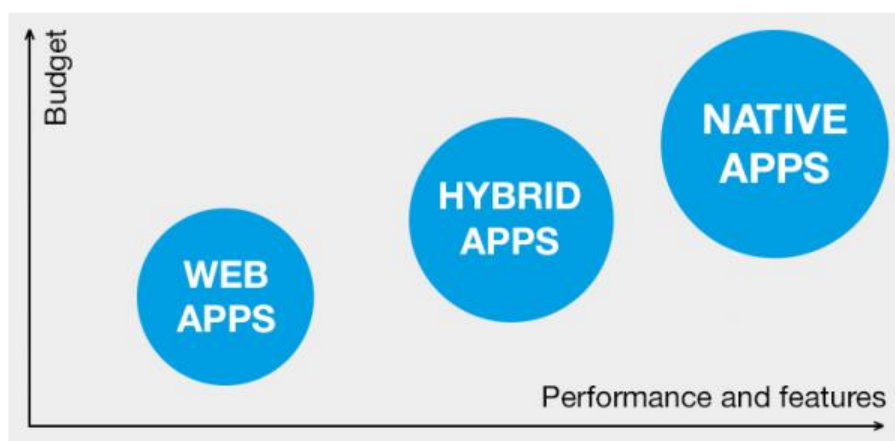
Hybridisovellus on yhdistelmä natiivisovellusta ja responsiivisesti toteutettua verkkosivustoa, jossa sovellukseen on upotettu internet-selain (Budi 2013). Karkeasti voitaisiin sanoa, että hybridisovellukset soveltuvat paremmin suunnittelijoille ja natiivisovellukset ovat sovelluskehittäjiä varten. Hybridisovellusten toteutuksessa ei tarvitse keskittyä yhtä paljon ohjelmointiin ja tekniseen puoleen. Esimerkiksi opinnäytetyönä toteutettuun iOS-hybridisovellukseen opinnäytetyön tekijä ei kirjoittanut yhtäkään riviä Objective-C-ohjelmointikieltä, jota käytetään iOS-sovellusten ohjelmoinnissa. Hybridisovelluksen etuna on myös se, että sovelluskehityksen kautta on helppo toteuttaa mobiilisovelluksesta omat versiot eri käyttöjärjestelmille, mikä säästää aikaa ja rahaa verrattuna usean natiivisovelluksen toteuttamiseen. (Layon 2011, 3.)

Kolmas vaihtoehto on web-sovellus, jonka nimi on harhaanjohtava, sillä se ei teknisesti ole sovellus ollenkaan. Web-sovellus on selaimessa toimiva mobiilioptimoitu verkkosivu, johon on toteutettu natiivisovellusta muistuttava käyttöliittymä. Natiivi- ja hybridisovellukset ladataan päätelaitteeseen käyttöjärjestelmän oman sovelluskaupan kautta omiksi sovelluksiksi, ja sen ansiosta niillä on mahdollisuus päästä käsiksi päätelaitteen natiiviominaisuuksiin, kuten kameraan ja paikannustietoihin, toisin kuin rajoitetummalla web-sovelluksella. Web-sovellus on näistä vaihtoehdoista helpoin toteuttaa, ja se toimii kaikissa laitteissa, joista löytyy internet-selain, mutta se on toisaalta myös kaikin rajoitetuin vaihtoehto. (Budi 2013.)

Natiivisovellus on vaihtoehdoista ainoa, joka ei perustu internet-sivuston käyttämiseen sovelluksen toteutuksessa, ja siksi sen avulla tehtyjä sovelluksia voi käyttää myös ilman internet-yhteyttä. Toisaalta natiivisovelluksen päivittäminen ei ole yhtä helppoa kuin

muiden vaihtoehtojen, sillä natiivisovelluksessa loppukäyttäjän täytyy aina ladata sovel-luskaupasta viimeisin päivitys sovellukseen, kun muissa vaihtoehdoissa päivitys voi-daan tehdä suoraan verkkosivustoon, jolloin päivitykset ovat heti voimassa. (Wroblews-ki 2011, 16.)

Mobiilisovelluksen toteutustekniikkaa valittaessa kannattaa siis kiinnittää huomiota toteutettavan mobiilisovelluksen ominaisuuksiin, tuettavien käyttöjärjestelmien määrään sekä sovelluksen tekemiseen varattuihin resursseihin. Kuva 9 havainnollistaa näiden kolmen toteutustekniikan erot käyttökokemuksen ja ominaisuuksien laajuuden sekä vaadittavan budjetin osalta.



KUVA 9. Gambit Group -sivuston artikkelissa verrataan toisiinsa eri toteutustekniikoi-la tehtyjen sovellusten käytettävyyttä ja ominaisuuksien laajuutta suhteessa vaadittavaan budjettiin (Wallin 2014).

## 4.2 Hybriditekniikka VirtualTampere-mobiilisovelluksen toteutuksessa

VirtualTampere-mobiilisovellus päätettiin toteuttaa hybriditekniikalla, koska mobiiliso-velluksen pääasiallisena tarkoituksena oli yksinkertaisesti avata päätelaitteen näytölle VirtualTampere-mobiilisivusto ilman selaimen omaa käyttöliittymää. Projektin toteut-tamiseen ei myöskään ollut käytettävissä natiivisovelluksen tekemiseen tarvittavaa oh-jelmointiosaamista. Hybridisovelluksessa kiinnosti myös mahdollisuus toteuttaa sovel-luskehysten kautta mobiilisovelluksesta helposti omat versiot eri käyttöjärjestelmille. (Mattus 2014.)



Hybridisovelluksen toteuttamiseen tarvittiin sovelluskehys, jonka avulla pystyttäisiin koostamaan iOS-mobiilisovellus, jossa VirtualTampere-mobiilisivusto olisi selattavissa. Sovelluskehystenä päätettiin käyttää kotimaista Neonto Studio Pro -ohjelmistoa, joka oli syksyllä 2014 vielä beta-asteella. Hybridisovellusten toteutukseen on olemassa muitakin sovelluskehyskiä, mutta Neonto Studio Pro -ohjelmistoon päädyttiin, koska toimeksiantaja halusi saada siitä käyttökokemuksia. (Mattus 2014.)

Neonto Studio Pro -sovelluskehys muodostaa Objective-C-kielellä iOS-mobiilisovellukselle valmiin rungon Xcode-ohjelmistoon. Xcode on Applen tarjoama ilmainen ohjelmointiympäristö, joka on saatavilla vain Mac-tietokoneille. iOS-sovelluksen toteutusta varten tarvitaan myös Applen maksullinen Developer ID. (Layon 2011, 16 - 23.)

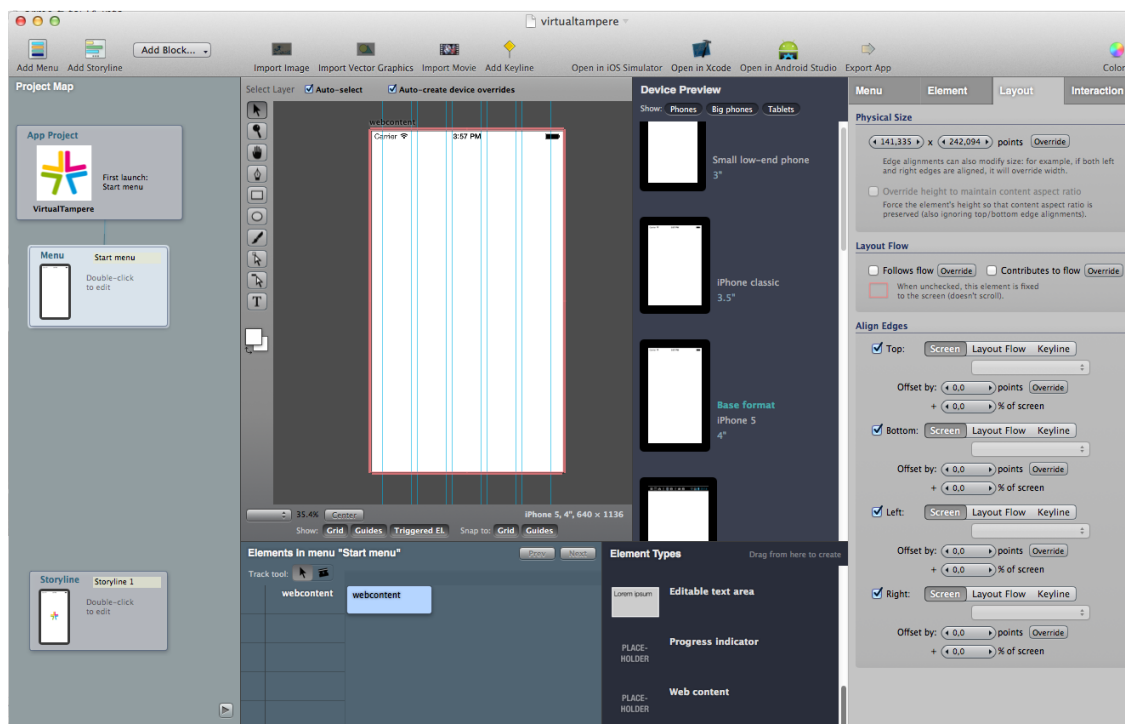
### **4.3 Sovelluksen tekninen toteutus**

Hybriditekniikkaan pohjautuvan mobiilisovelluksen tekeminen Neonto Studio Pro -ohjelmistolla osoittautui yksinkertaiseksi. Aluksi luotiin mobiilisovellusprojekti ja määriteltiin, onko se tarkoitettu käytettäväksi älypuhelimella, tablet-tietokoneella vai molemmilla. Tämän jälkeen pudotusvalikosta valittiin kategoria, joka kuvastaa sovellusta parhaiten, ja ladattiin projektiin mobiilisovelluksen ikonina käytettävä kuva.

Seuraavaksi luotiin ensimmäinen, ja tämän projektin ainoa, näkymäsivu sovellukselle. Neonto Studio Pro -ohjelmistosta olisi löytynyt monia työkaluja esimerkiksi käyttöliittymän toteuttamiseen, mutta koska tässä projektissa tarvittiin vain yksi koko näytön kokoinen selainnäkyvä, valittiin tarjolla olleista vaihtoehdoista ”web content” -elementti. Se raahattiin elementtivalikosta esikatselunäkymän päälle, ja tämän jälkeen päästiin käsiksi valitun elementin ominaisuuksien määrittelyihin.

”Web content” -elementin ominaisuuksissa määriteltiin ensimmäiseksi verkkosivuosoitte, joka siihen avataan sovelluksen käynnistyessä. Osoitteeksi laitettiin <https://www.virtualtampere.com>, joka myöhemmin vaihtui vielä muotoon <https://www.virtualtampere.com/?appview>. Seuraavaksi elementin ulkoasu määriteltiin sellaiseksi, että se asettuisi koko päätelaitteen näytön levyiseksi ja korkuiseksi riippumatta laitteen näytön koosta. Tämä onnistui määrittämällä erikseen elementin jokaisen

sivun ja laitteen näytön sitä vastaavan ulkoreunan väliseksi etäisyydeksi 0. ”Web content” -elementille ei siis määritelty kiinteää leveyttä ja korkeutta, vaan se mukautui aina kunkin päätelaitteen näytön kokoiseksi. Kuvakaappaus Neonto Studio Pro -ohjelmistosta ja elementin asettelun määrittelyistä on nähtävissä kuvassa 10.

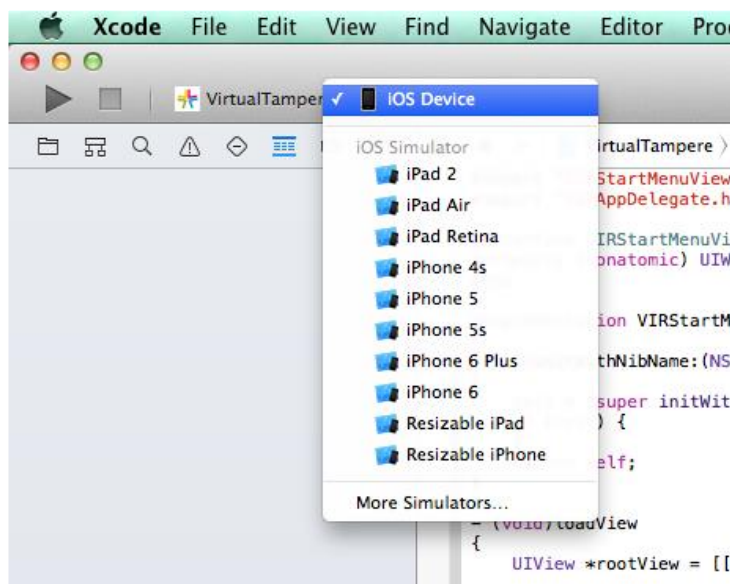


KUVA 10. Kuvakaappaus VirtualTampere-projektista Neonto Studio Pro -ohjelmistossa.

Tämä työvaihe saatiin näiden asetusten myötä valmiiksi ja projekti oli valmis koostettavaksi iOS-sovellusprojektiksi. Se onnistui todella helposti klikkaamalla Neonto Studio Pro -ohjelmiston yläreunasta ”Open in Xcode” -painiketta, ja vain muutamaa sekuntia myöhemmin Xcode-ohjelmistoon oli luotu täysin toimiva, Objective-C-ohjelmointikieltä käyttävä iOS-sovellus.

#### 4.4 Sovelluksen testaus

Testaaminen Xcode-ohjelmistossa onnistui klikkaamalla vasemmassa yläreunassa näkyvää ”play”-painiketta. Testauksessa käytettiin pääasiassa iPhone 5S -älypuhelinia, jonka Xcode tunnisti automaattisesti, kun laite liitettiin tietokoneen USB-porttiin. Xcode-ohjelmistosta löytyi testausta varten myös virtuaalisia iPhone-simulaattoreita. Valikoima Xcode-ohjelmistossa valittavista testauslaitteista on nähtävissä kuvassa 11.



Kuva 11. Kuvakaappaus valittavista testilaitteista Xcode-ohjelmistossa.

Sovelluskehiksen avulla koostettu iOS-sovellus käynnistyi heti ensi yrittämällä ja mobiilisivuston kaikki eri ominaisuudet toimivat. Sovelluksessa huomattiin kuitenkin tarvetta hienosäädölle:

- Sovellukseen upotettu selainikkuna toimi, mutta sen asettelussa oli vikaa, sillä verkkosivun alareunasta osa jäi pois näkyvistä.
- iPhoneen ”status bar” näkyi sovelluksessa selainikkunan päällä, vaikka tarkoitus oli saada koko näyttö ainoastaan sovelluksen käyttöön.
- Sovelluksen alareunassa olevat linkit JJ-Net Group Oy:n kotisivuille ja Facebookiin osoittautuivat tarpeettomiksi, koska toiselle sivustolle siirryttyä käyttäjällä ei ollut mahdollisuutta palata enää takaisin alkuperäiselle sivulle, koska selaimen ”edellinen sivu”-painike puuttui.
- Mobiilisovellusta varten mobiilisivustolle piti tehdä lisämuutoksia, jotta sovelluksen käyttökokemus muistuttaisi enemmän natiivisovellusta.

Ensimmäisiä muutoksia varten alettiin käydä läpi sovellusprojektin tiedostorakennetta ja löydettiin ”VIRStartMenuViewController.m”-tiedosto. Sitä muokkaamalla iPhoneen ”status bar” pystyttiin piilottamaan ja sovellukseen upotettu selainikkuna saatiin aseteltua laitteen näytölle halutulla tavalla. Muut muutokset vaativat hienosäätöä itse mobiilisivustoon, joten upotuksessa käytetyn URL-osoitteen loppuun päätettiin lisätä kappaleessa 4.3. mainittu ”?appview”-tunniste. Tämän tunnisteiden avulla tehtiin muutoksia CSS-tyylimäärittelyihin sekä jQuery-toimintoihin, jotka vaikuttivat ainoastaan mobiilisovelluksen kautta näytettävään versioon VirtualTampereen-mobiilisivustosta. ”?app-

view”-tunnistetta hyödyntämällä saatiin verkkosivuston alapalkki piilotettua mobiilisovelluksesta ja mobiilisovelluksen käytettävyyttä saatiin hiottua muistuttamaan enemmän natiivisovellusta.

Sovelluksen testauksessa käytettiin iPhone 4S- ja iPhone 5S -puhelinten lisäksi Xcode-ohjelmiston iPhone-simulaattoreita. Wroblewski (2011, 120) kehottaa käyttämään testauksessa oikeita älypuhelimia aina kun mahdollista, sen sijaan että käyttäisi simulaattoria tai pöytäkoneen selainta. VirtualTampere-sovelluksen testauksesta Xcoden iPhone-simulaattori kuitenkin suoriutui virheettömästi, toistaen kaikki bugit samalla tavalla kuin oikeat iPhonet.

## 5 POHDINTA

Opinnäytetyönä tehtyjen mobiilisivuston sekä mobiilisovelluksen toteutus onnistui alussa asetettujen tavoitteiden mukaisesti. Opinnäytetyön tekijä sekä toimeksiantaja olivat tyytyväisiä lopputuloksiin, joten työtä voidaan pitää onnistuneena. Opinnäytetyö antaa-kin perustiedot hybriditekniikalla tehdyn sovelluksen toteuttamisesta ja siihen liittyvistä työvaiheista.

Opinnäytetyössä käytiin läpi eri sovellustekniikkojen eroja, ja niiden avulla voidaan päätellä, miksi toteutetussa sovelluksessa päätettiin käyttää hybriditekniikkaa. Opinnäytetyö oli aiheeltaan melko laaja, joten joistakin osa-alueista mahtui mukaan vain pintaraapaisu. Opinnäytetyössä käytiin läpi monia mobiilisovellukseen kuuluvia työvaihteita ja niistä olisikin pystynyt kirjoittamaan useita vain yhtä työvaihetta syvällisemmin kuvaavia opinnäytetöitä.

Tulevaisuudessa mobiilisovellusta jatkokehitetään vielä tekemällä käyttäjätestausta, jonka avulla sovelluksen käytettävyyttä pyritään tekemään intuitiivisemmaksi. Opinnäytetyön tekemisen aikana Neonto Studio Pro -sovelluskehikseen julkaistiin päivitys, jonka avulla pystyy koostamaan mobiilisovelluksen myös Android-käyttöjärjestelmälle Android Studio -ohjelmointiympäristöön. Sitä tullaankin kokeilemaan mobiilisovelluksen jatkokehityksessä. Opinnäytetyössä ei käsitelty tehdyn mobiilisovelluksen julkaisua sovelluskaupassa, koska se tullaan julkaisemaan vasta myöhemmin, kun sille on tehty jatkokehitystä ja testausta.

## LÄHTEET

Budiu, R. 2013. Mobile: Native App, Web Apps, and Hybrid Apps. Nielsen Norman Group. Luettu 27.9.2014. <http://www.nngroup.com/articles/mobile-native-apps/>

Firtman, M. 2010. Programming the Mobile Web. Sebastopol: O'Reilly Media, Inc.

Historia. 2014. JJ-Net Group Oy. Luettu 19.9.2014. <http://www.jj-net.fi/yritys/historia>

Jeskanen, J. toimitusjohtaja. 2014. Haastattelu 17.09.2014. Haastattelija Järvensivu, J. Tampere.

JJ-Net Group Oy. 2014. JJ-Net Group Oy. Luettu 19.9.2014. <http://www.jj-net.fi/yritys>

Layon, K. 2011. The Web Deigners Guide to iOS Apps. Berkeley: New Riders

Marcotte, E. 2011. Reponsive web design. A book apart. New York: A Book Apart.

Mattus, J. sovelluskehittäjä. 2014. Haastattelu 10.10.2014. Haastattelija Järvensivu, J. Tampere.

O'Toole, J. 2014. Mobile apps overtake PC Internet usage in U.S. Luettu 13.11.2014. <http://money.cnn.com/2014/02/28/technology/mobile/mobile-apps-internet/>

Pilgrim, M. 2014. No 11. Manipulating history for fun & profit. Luettu 24.10.2014. <http://diveintohtml5.info/history.html>

Street View Service. 2014. Google Developers. Luettu 23.10.2014. <https://developers.google.com/maps/documentation/javascript/images/panoramaSphere.png>

Vanha kirjastotalo. Kuvakaappaus panoraamaesityksestä. VirtualTampere 2014. Katsottu 29.9.2014. <https://www.virtuالتampere.com/vanha-kirjastotalo>

Virtual Tampere. 2014. Virtual Tampere. Luettu 19.9.2014. <https://www.virtuالتampere.com>

Wallin, T. 2014. Verkko, hybridi tai natiivi. Luettu 19.9.2014. <http://www.gambitgroup.fi/projektimme/verkko-hybridi-tai-natiivi/>